Breakpoint App Note

There are many types of breakpoints implemented in the American Arium WinDb software: Execution, Memory Access, I/O Access, etc. Each of these types is implemented in one or more of the following resource categories: Processor, Software, Bus Analyzer, and Emulator. This "resource" categorization of the breakpoints is based upon how the breakpoint is implemented. Each category has its advantages and disadvantages over one another. The following is a brief discussion of these issues.

All the breakpoint resources, with the exception of the Bus Analyzer category, are available on all models of the American Arium emulators. The Bus Analyzer breakpoints are available only on the TRC-55 and TRC-6 products because these products include real-time bus tracing needed to implement this breakpoint resource.

Processor breakpoints are implemented using on-chip resources of the target processor such as the Debug Registers. These on-chip resources were designed into the processor for the purpose of supporting both native debuggers (such as SoftICE) and In-Circuit Emulators like the American Arium product line. This is the default resource used for code execution breakpoints.

Software breakpoints use a method called instructions substitution. This limits them to code execution breaks and also limit them to RAM or other writable memory, but does allow for a larger number of breakpoints than any other resource category.

Emulator breakpoints are a category of breakpoints that are implemented via emulator hardware (for example, the Reset breakpoint).

Pros/Cons of Software, Processor, Bus Analyzer, and Emulator breakpoints
SOFTWARE
Pro:
- Up to 64 breakpoints active at any time.
- Stops processor before instruction is executed.
- Not processor-specific in an SMP target.

Con:
- Can’t break on ROM code. The instruction substition method requires writable memory in order to work.
- Limited to code execution breaks. Can't do memory access or I/O access breaks.
- Should not be used when paging is enabled.
- Self-modifying code will prevent the breakpoint from working.

PROCESSOR
Pro:
- Stops processor before instruction is executed.
- Can break on code execution, memory access, or I/O access.

Con:
- Only 4 available per processor. This limit is imposed by the processor architecture.
- Processor overhead in some cases (target runs slower than full speed).
- Processor-specific in an SMP target. Must explicitly set the breakpoint in each processor if a "global" breakpoint is desired.
- Code on the target can modify the debug registers, preventing the breakpoint from working.

BUS ANALYZER

Pro:

- Breakpoints affect all processors in an SMP system.
- Non-intrusive (processor runs full speed).
- Break on events other than memory and I/O cycles (IACK, BTM, Shutdown, etc.).
- Break on sequence of events: sequences up to 4 levels deep.
- Break on count of events.
- Data value can be specified in addition to an address.

Con:

- Processor executes slightly past the break event (slide).
- Can't break on code execution (can break on Fetch).
- Not effective on cached accesses.
- Many fetches and memory operations occur on cache-line sized operations. This can make it difficult to trigger exclusively on the desired event.

EMULATOR

Emulator breakpoints are distinct from all the other breakpoint types and therefore a list of pro's and con's doesn't readily apply in this context.

Conclusion:

Though this categorization may seem complex, the choices that need to be made when adding a breakpoint are simplified by the fact that most breakpoint types are implemented in only one or two resource categories. When a choice of the resource is available, the decision of which resource to use depends on the situation at hand. Weighing the pro's and con's listed above will help decide which resource is the most appropriate for the situation. When in doubt, multiple breakpoints of the same type using different resources can be used. As with any tool, experience is the best teacher.

14811 Myford Road
Tustin, CA 92780
Voice: 714-731-1661
Fax: 714-731-6344
Web: www.arium.com
E-mail: info@arium.com